SPECIFICATION

Please amend the specification with the following replacement paragraphs numbered according to the published application:

[0017] g) ~~and~~ rejection of candidate signatures with overall scores which fail to achieve the threshold.

[0032] e) a .DLL ~~.DDL~~ or .ocx file; and

[0046] Referring to FIG. 1, a flow diagram of a steganography detection process 10 of the invention is shown, which is implemented by a computer system (not shown). The computer system starts the process 10 at 12 and obtains a steganographic signature directory at 14 as will be described in more detail later. At 16, a check is made to see if there is any unread file in the directory: if there is such a file, at 18 a sample of length N bytes is read from a prearranged location in the file into a steganographic signature array directory. Here N is a positive integer chosen to be sufficiently large to avoid false positives as far as practically possible, and sufficiently small to allow the process 10 to detect use of variants of a steganographic program. In this example N was 500. In this example the prearranged location of the N byte sample is the beginning of the file. The computer system then iterates around steps 16 and 18 via a loop 20 until all files in the steganographic signature directory have been read and have provided respective N byte samples for the steganographic signature array directory. Each N byte sample of program code is used without alteration as the signature for a respective steganographic program. This is a particularly simple way of obtaining signatures: it is not necessary to process the files or extract ~~extracts~~ from them in any way other than to read part of each file. It is not essential to read from the start of each file, N successive bytes can be read from anywhere in a file. However, bytes at the beginning of a file are more convenient because they are less likely to change in compilation. It is also possible to use more than one sample from a file, and to have different sample lengths for different files, albeit these options are less convenient.

[0051] It is straightforward to obtain copies of most or all publicly available steganographic programs, as they are available free or obtainable by purchase, often from the Internet. In order to derive steganographic signatures, an important file in each of these steganographic programs was identified and chosen: wherever possible, this chosen file was the program's core steganographic kernel. For example, a steganographic program may comprise an ".exe" (execute) file to provide a computer user interface, and a ".DLL" (dynamically linked library) file to perform a computationally intensive computation when called by the .exe file. For such a program, the .DLL file was chosen to provide the N byte sample at 18 as it is liable to be smaller than the .exe file and more importantly less liable to be changed. A .DLL file is often a file which implements a mathematical function, and its kernel is that part of it which is essential to implement that function. Choosing a .DLL file also means that attempts to write a new steganographic program using it would also be detected. It is best if the N byte sample is completely unique to the steganographic program to avoid false positives, and a .DLL file is more likely to be unique than an .exe file. Another possibility is a .ocx file which is similar to a .DLL ~~.DDL~~ file except that it has a different interface.